Eric Spencer

(630) 796 5368 | ericspencer1450@gmail.com | linkedin.com/in/ericspencer00

EDUCATION

Loyola University Chicago *expected May 2026*

BS/BA in **Computer Science**, Minor in **Information Systems**

Dean's List Fall 2023, Trustee Scholarship

SKILLS

Advanced Proficiency in Java, LaTeX, and the Linux Terminal.

Experience with Git Control, C, C++, Python, x86 Assembly, AI Prompt Engineering, RStudio, Microsoft Excel and HTML/CSS.

RELEVANT COURSE WORK

Data Structures I & II (Java), Computing Tools (Linux), Computer Systems (C), Discrete Structures, Calculus I & II, and Probabilities and Statistics (RStudio)

EXPERIENCE

Unchained AI **Chicago IL**

Junior AI Prompt Engineer *January 2024 – Present*

- Solve problems for customers' workflows using Open-AI's Chat-GPT v4.0's image and text processing.
- Experiment with different prompts to produce the most optimal output.

Loyola University Chicago **Chicago IL**

Research Assistant *May 2023 – August 2023*

- Played a pivotal role in studying how Chat-GPT v3.0 understood Pearson CS2 Java Data Structures and collaborated with fellow computer science professors and students.

Oak Park Country Club **River Grove, IL**

Honor Caddie *Summers 2017 - 2023*

- Engaged with up to 4 members and fellow caddies at once over a 4-hour golf session, multiple times a day to aid in their game and well-being.
- Promoted to an honor caddie within 2 years through experience and advanced knowledge in the game of golf and caddying protocol.

PERSONAL PROJECTS

Anagram Solver for Scrabble

- Designed and implemented a solver for a set of letters that processes a dictionary to generate the permutations of all possible playable words that incorporates wildcards.
- Utilized HashSets and HashMaps to enhance word lookup speed and manage wildcard options effectively.

Stack vs Recursive Maze

- Conducted a comparative study of maze-solving algorithms, showcasing the performance trade-offs between recursive and stack-based approaches.

FreeTime Calculator

- Developed a user-friendly application to calculate and display free time slots within a specified day. Through loop structures and array manipulation, the program parses and compares schedules to output a cumulative free time for all users.